PORTADA

ALUMNO:JOSE FABIAN HERNANDZ MARROQUIN

GRUPO:"A"

GRADO:2°

PROFESOR: CARLOS ALBERTO GONZALES GONZALES

TEMA: TRIGGERS

¿QUE ES UN TRIGGER?

Un ***trigger*** (o disparador) en una Base de datos , es un procedimiento que se ejecuta cuando se cumple una condición establecida al realizar una operación. Dependiendo de la base de datos, los triggers pueden ser de inserción (INSERT), actualización (UPDATE) o borrado (DELETE). Algunas bases de datos pueden ejecutar triggers al crear, borrar o editar usuarios, tablas, bases de datos u otros objetos.

CARACTERISTICAS

- **Llamada de activación**: es la sentencia que permite "disparar" el código a ejecutar.
- **Restricción**: es la condición necesaria para realizar el código. Esta restricción puede ser de tipo condicional o de tipo nulidad.
- **Acción a ejecutar**: es la secuencia de instrucciones a ejecutar una vez que se han cumplido las condiciones iniciales.

VENTAJAS Y DESVENTAJAS

- **DESVENTAJAS**
- No aceptan parámetros o argumentos (pero podrían almacenar los datos afectados en tablas temporales)
- No pueden ejecutar las operaciones *COMMIT* o *ROLLBACK* por que estas son parte de la sentencia SQL del disparador (únicamente a través de transacciones autónomas)
- Pueden causar errores de mutaciones en las tablas, si se han escrito de manera deficient
- **VENTAJAS**
- Los disparadores son soportados en MySQL a partir de la versión 5.0.2

•Fuerzan restricciones dinámicas de integridad de datos y de integridad referencial.

•Aseguran que las operaciones relacionadas se realizan juntas de forma implícita.

•Respuesta instantánea ante un evento auditado

•Ofrece un mayor control sobre la B.D

MODO DE EMPLEO

Son usados para mejorar la administración de la Base de datos, sin necesidad de contar con que el usuario ejecute la sentencia de SQL.

Además, pueden generar valores de columnas, previene errores de datos, sincroniza tablas, modifica valores de una vista, etc.

Permite implementar programas basados en paradigma lógico (sistemas expertos, deducción).

3 EJEMPLOS DE TRIGGERS

```
ALTER TRIGGER TR_CUENTAS
ON CUENTAS
AFTER UPDATE
AS
BEGIN
-- SET NOCOUNT ON impide que se generen mensajes de texto
-- con cada instrucción
SET NOCOUNT ON;

IF UPDATE(SALDO) -- Solo si se actualiza SALDO
BEGIN
INSERT INTO HCO_SALDOS
(IDCUENTA, SALDO, FXSALDO)
SELECT IDCUENTA, SALDO, getdate()
FROM INSERTED
END
END
```

Ejemplo 2

```sql
CREATE TRIGGER TR_CUENTAS

ON CUENTAS

AFTER UPDATE

AS

BEGIN

-- SET NOCOUNT ON impide que se generen mensajes de texto

-- con cada instrucción

SET NOCOUNT ON;

INSERT INTO HCO_SALDOS

(IDCUENTA, SALDO, FXSALDO)

SELECT IDCUENTA, SALDO, getdate()

FROM INSERTED

END
```

Ejemplo 3

```
CREATE TRIGGER TR_SEGURIDAD

ON DATABASE FOR DROP_TABLE, ALTER_TABLE

AS

BEGIN

RAISERROR ('No está permitido borrar ni modificar tablas !' , 16, 1)

ROLLBACK TRANSACTION

END
```